

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-093169

(43)Date of publication of application : 07.04.1995

(51)Int.Cl.

G06F 9/46

(21)Application number : 05-239305

(71)Applicant : PFU LTD

(22)Date of filing : 27.09.1993

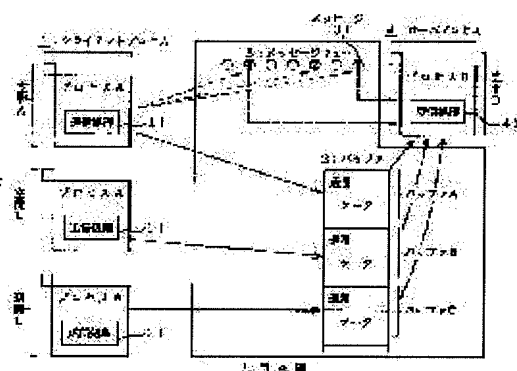
(72)Inventor : KOMORI TAKAKAZU

## (54) INTER-PROCESS COMMUNICATION SYSTEM

## (57)Abstract:

PURPOSE: To perform the processing of requested data in real time and to quickly return a state to the one to process the data in real time by interrupting futile processing even when a large number of messages reside due to an unknown factor in a message queue.

CONSTITUTION: This system is constituted in such a way that a buffer 2 provided at every client process 1 in which each client process 1 sets data and a consecutive number, and the message queue to connect the leading address of the data overwritten on the buffer 2 to the message 31 on which the consecutive number is set are provided at shared space, and processing to fetch the data from the buffer 2 is performed only when the consecutive number of the message 31 taken out from the forefront of the message queue 3 by a server process 4 is not equal to that of the data in the buffer 2 at the leading address, and the next processing is started by interrupting the processing when noncoincidence is obtained between them.



(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平7-93169

(43)公開日 平成7年(1995)4月7日

(51)Int.Cl.<sup>6</sup>

G 0 6 F 9/46

識別記号

3 4 0 B 8120-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数 3 O L (全 8 頁)

(21)出願番号 特願平5-239305

(22)出願日 平成5年(1993)9月27日

(71)出願人 000136136

株式会社ピーエフユー

石川県河北郡宇ノ気町宇野気ヌ98番地の  
2

(72)発明者 小森 尊和

神奈川県大和市深見西四丁目2番49号 株  
式会社ピーエフユー大和工場内

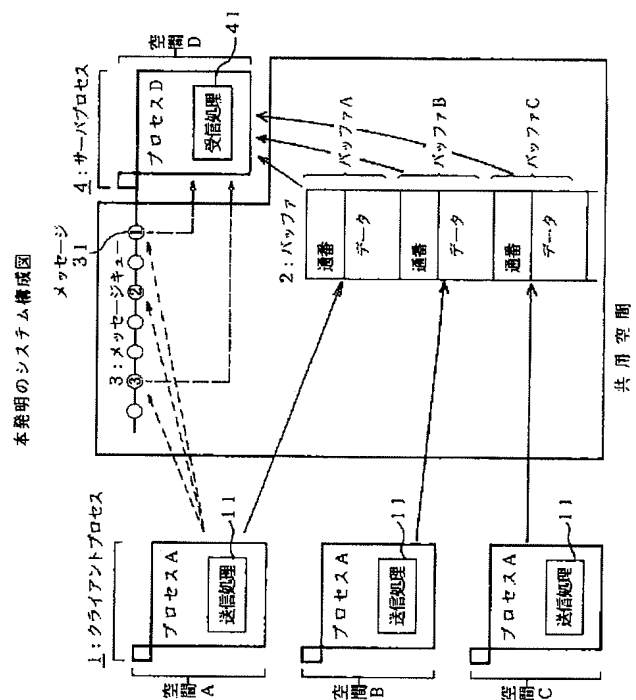
(74)代理人 弁理士 岡田 守弘

(54)【発明の名称】 プロセス間通信方式

(57)【要約】

【目的】 本発明は、プロセス間通信方式に関し、依頼を受けたデータの処理をリアルタイムに行なうと共にたとえメッセージキューに多くのメッセージが何らかの原因で滞留しても無駄な処理を中止して速やかにデータをリアルタイムに処理する状態に戻すことを目的とする。

【構成】 各クライアントプロセス1がデータおよび通番を設定する各クライアントプロセス1毎に設けたバッファ2と、バッファ2に上書きしたデータの先頭アドレスおよび通番を設定したメッセージ31をつなぐメッセージキュー3とを共用空間に設け、サーバプロセス4がメッセージキュー3の先頭から取り出したメッセージ31の通番と、先頭アドレスのバッファ2内のデータの通番とが等しい場合にのみ当該バッファ2からデータを取り出して処理を行ない、等しくない場合に処理中止して次の処理に進むように構成する。



1

## 【特許請求の範囲】

【請求項1】 プロセス間で通信を行なうプロセス間通信方式において、

処理依頼する各クライアントプロセス（1）がデータを上書きするおよび当該データを上書きする毎に変えた通番を設定する各クライアントプロセス（1）毎に設けたバッファ（2）と、

このバッファ（2）に上書きしたデータの先頭アドレスおよび上記通番を設定したメッセージ（31）をつなぐメッセージキュー（3）とを共用空間に設け、  
依頼された処理を行なうサーバプロセス（4）が上記メッセージキュー（3）の先頭から取り出したメッセージ（31）の通番と、設定されていた先頭アドレスのバッファ（2）内のデータの通番とが等しい場合にのみ当該バッファ（2）からデータを取り出して処理を行ない、等しくない場合に処理中止して次の処理に進むように構成したことを特徴とするプロセス間通信方式。

【請求項2】 請求項1において、バッファ（2）の先頭アドレスが固定の場合に、上記通番を設定したメッセージ（31）をメッセージキュー（3）につなぐように構成したことを特徴とするプロセス間通信方式。

【請求項3】 上記クライアントプロセス（1）は複数のプロセスからなり、これら各プロセスが自プロセスIDと通番、更に必要に応じて先頭アドレスを設定したメッセージ（31）を上記メッセージキュー（3）につなぐように構成したことを特徴とする請求項1および請求項2記載のプロセス間通信方式。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、プロセス間で通信を行なうプロセス間通信方式に関するものである。

## 【0002】

【従来の技術】 従来、複数のプロセスからある1つのプロセスにデータを非同期に渡して処理依頼する場合、図7に示すように、共用空間内にメッセージをつなぐメッセージキューと、処理依頼するデータを書き込んで渡すためのバッファを設け、各プロセスが非同期に当該バッファにデータを書き込むと共にメッセージをメッセージキューにつないで処理依頼する。処理を行なうプロセスは、メッセージキューの先頭からメッセージを取り出して該当するバッファからデータを読み出し、処理を行なうことを繰り返す。以下図7の構成および動作を簡単に説明する。

【0003】 図7は、従来技術の説明図を示す。図7において、プロセスA、B、Cは、非同期にデータの処理依頼を行なう空間A、B、C内のそれぞれのプロセスである。

【0004】 バッファA、B、Cは、共用空間内に設け、プロセスA、B、Cがそれぞれデータを書き込むために割り当てたそれぞれのバッファである。このバッ

2

ファA、B、Cには、プロセスA、B、Cがそれぞれ最新のデータを非同期に上書きする。

【0005】 プロセスDは、プロセスA、B、Cから処理依頼を受けたデータをバッファA、B、Cからそれぞれ読み出して処理（例えば表示処理）を行なうものであって、空間Dに設けたものである。

【0006】 メッセージキューは、プロセスA、B、Cがそれぞれ処理依頼する内容およびデータを書き込むバッファA、B、Cの先頭アドレスを設定したメッセージをつないで、プロセスDに処理依頼するためのものであって、共用空間に設けたものである。プロセスDは、当該メッセージキューの先頭から順次メッセージを取り出し、処理を行なう。

【0007】 バッファA、B、Cは、プロセスA、B、Cがそれぞれデータを上書きする領域であって、共用空間内に設けたものである。次に、動作を説明する。

【0008】 （1） プロセスAが処理依頼するデータをバッファAに書き込むと共に自プロセスAおよびバッファAの先頭アドレスを設定した処理依頼をメッセージに書き込んでメッセージキューにつなぐことを非同期に行なう。同様に、他のプロセスB、CもデータをバッファB、Cに書き込むと共に自プロセスB、CおよびバッファB、Cの先頭アドレスをそれぞれ設定した処理依頼をメッセージに書き込んでメッセージキューにつなぐことを非同期にそれぞれ行なう。

【0009】 （2） プロセスDがメッセージキューの先頭から1つメッセージを取り出し、当該メッセージで指示された先頭アドレスのバッファA、B、Cのいずれからデータを取り出し、メッセージで指示された処理（例えば表示処理）を行なうことを繰り返す。

【0010】 以上のように、プロセスA、B、Cが非同期にデータをバッファA、B、Cに書き込み、プロセスDがバッファA、B、Cから順次データを取り出して処理（例えば表示処理）を行なうようにしていた。

## 【0011】

【発明が解決しようとする課題】 上述した図7の構成によるプロセスA、B、Cが非同期にデータをバッファA、B、Cに書き込み、プロセスDが順番に取り出して処理を行っていたのでは、プロセスDのデータ処理が何らかの原因により遅れた場合（例えば外部I/Oなどのアクセス待ちにより処理が遅れた場合など）、メッセージキューにつながれたメッセージを取り出して処理を行なうことが遅れてしまい、プロセスA、B、Cから非同期に送られてきたバッファA、B、C上のデータをリアルタイムに処理できない、例えば画面上に現時刻のデータをリアルタイムに表示できないという問題が発生した。また、プロセスDの処理が一度遅れてメッセージキューにメッセージが数多く滞留してしまうと、メッセージキューにつながれたメッセージの数がなかなか減少しなくて当該プロセスDの負荷が大の時間が長時間続いて

しまい他の処理を行えないと共に、リアルタイムに処理できるようになるまでに多くの時間が必要になってしまいう問題もあった。

【0012】本発明は、これらの問題を解決するため、各クライアントプロセスがバッファにデータと通番を書き込むと共にメッセージにも通番を設定してメッセージキューにつないで処理依頼し、サーバプロセスがメッセージキューから取り出したメッセージの通番とバッファ上のデータの通番とが等しい場合にのみ処理を行い、等しくないときに処理中止して最新のデータのみ処理を行ない、依頼を受けたデータの処理をリアルタイムに行なうと共にたとえメッセージキューに多くのメッセージが何らかの原因で滞留しても無駄な処理を中止して速やかにデータをリアルタイムに処理する状態に戻すことを目的としている。

【0013】

【課題を解決するための手段】図1を参照して課題を解決するための手段を説明する。図1において、クライアントプロセス1は、データの処理依頼するものである。

【0014】バッファ2は、共用空間に設け、各クライアントプロセス1がデータを書き込む領域である。メッセージ31は、処理依頼するデータのバッファ2上の先頭アドレスおよびデータの通番を設定するものである。

【0015】メッセージキュー3は、メッセージ31をつないで、サーバプロセス4に処理依頼するためのものである。サーバプロセス4は、メッセージキュー3の先頭からメッセージ31を順次取り出し、当該メッセージ31に設定されていた通番と、設定されていた先頭アドレスのバッファ2のデータの通番が等しい場合にのみ処理を行なうものである。

【0016】

【作用】本発明は、図1に示すように、各クライアントプロセス1がバッファ2にデータおよび当該データの通番を書き込むと共に、当該バッファ2の先頭アドレスおよびデータの通番を設定したメッセージ31をメッセージキュー3につなぎ、サーバプロセス4がメッセージキュー3の先頭からメッセージ31を順次取り出し、この取り出したメッセージ31内の通番と、先頭アドレスのバッファ2内のデータの通番とが等しい場合にのみバッファ2からデータを取り出して処理を行ない、等しくない場合に処理中止して次の処理に進むようにしている。

【0017】また、バッファ2の先頭アドレスが固定の場合に、クライアントプロセス1が通番を設定したメッセージ31をメッセージキュー3につなぎ、サーバプロセス4がメッセージ31から取り出した通番と、固定の先頭アドレスのバッファ2内のデータの通番とが等しい場合にのみバッファ2からデータを取り出して処理を行ない、等しくない場合に処理中止して次の処理に進むようにしている。

【0018】また、クライアントプロセス1として複数

のプロセスを設け、これら各プロセスが自プロセスIDと通番、更に必要に応じて先頭アドレスを設定したメッセージ31をメッセージキュー3につなぐようにしている。

【0019】従って、各クライアントプロセス1がバッファ2にデータと通番を書き込むと共にメッセージ31にも通番を設定してメッセージキュー3につないで処理依頼し、サーバプロセス4がメッセージキュー3から取り出したメッセージ31の通番とバッファ2上のデータの通番とが等しい場合にのみ処理を行い、等しくないときに処理中止して最新のデータのみ処理を行なうことにより、依頼を受けたデータの処理をリアルタイムに行なうと共にたとえメッセージキュー3に多くのメッセージ31が何らかの原因で滞留しても無駄な処理を中止して速やかにデータをリアルタイムに処理する状態に戻すことが可能となる。

【0020】

【実施例】次に、図1から図6を用いて本発明の実施例の構成および動作を順次詳細に説明する。

【0021】図1は、本発明のシステム構成図を示す。図1において、クライアントプロセス1は、データの処理をサーバプロセス4に非同期に依頼するものであって、ここでは、例えば図示のようにプロセスA、プロセスB、プロセスCなどであり、それぞれ空間A、空間B、空間Cなどと別空間に存在するものである。クライアントプロセス1であるプロセスA、プロセスB、プロセスCはそれぞれ送信処理11などから構成されるものである。

【0022】送信処理11は、データおよび当該データの通番をバッファ2に書き込んだり、あるいはデータ変化に関係なく新たな通番をバッファ2に書き込んだりし、データの通番およびバッファ2の先頭アドレスなどを設定したメッセージ31をメッセージキュー3につないだりなどするものである。

【0023】バッファ2は、共用空間に設けたものであって、クライアントプロセス1であるプロセスA、プロセスB、プロセスCがデータおよび通番を書き込むものである。ここでは、プロセスA、プロセスB、プロセスC用のバッファ2として固定先頭アドレスを持つバッファA、バッファB、バッファCを設けている。これらバッファA、バッファB、バッファCには、図示のように、通番およびデータを上書き、常に最新の通番およびデータを書き込むようにしている。

【0024】メッセージキュー3は、バッファ2の先頭アドレスおよび通番を設定したメッセージ31を非同期につなぎ、処理順を管理するものである。サーバプロセス4は、メッセージキュー3の先頭からメッセージ31を順次取り出して処理を行なうものであって、受信処理41などから構成されるものである。

【0025】受信処理41は、メッセージキュー3の先

5

頭からメッセージ31を取り出し、この取り出したメッセージ31に設定されていた通番と、設定されていた先頭アドレスのバッファ2のデータの通番が等しい場合にのみ処理（例えば画面上にデータを表示する処理）を行なうものである。従って、バッファ2内のデータが上書きされているか否かに関係なく、通番が等しい最新のデータを当該バッファ2から取り出して処理を行なうものである。

【0026】以下順次詳細に説明する。図2は、本発明のクライアントプロセスの送信処理フローチャートを示す。図2において、S1は、バッファ2にデータをセットする。これは、図1のクライアントプロセス1である、例えばプロセスAの送信処理11が、サーバプロセス4に処理依頼する対象のデータを、バッファ2内の当該プロセスA用のバッファAに上書きする（セットする）。

【0027】S2は、バッファ2にデータの通番をセットする。これは、S1で、例えばプロセスAの送信処理11がバッファAにデータをセットしたことに続いて、当該バッファAにデータの通番を上書きする（セットする）。

【0028】S3は、メッセージ作成（データ処理要求）する。これは、例えばプロセスAが、後述する図4のメッセージ31に示すように、データ処理要求（種別）、A（送信元プロセスID）、バッファの先頭アドレス、およびデータの通番を設定したメッセージ31を作成する。

【0029】S4は、メッセージ送信する（メッセージキュー3につなぐ）。これは、S3で作成したメッセージ31を、メッセージキュー3の末尾につなぐ（エンキューする）。これにより、メッセージ31を送信依頼したこととなる。

【0030】以上によって、クライアントプロセス1において、送信処理11が処理依頼しようとするデータおよび当該データの通番をバッファ2に上書きした後、当該バッファ2の先頭アドレスおよびデータの通番をメッセージ31に設定してメッセージキュー3の末尾につなぎ、送信依頼する。これらにより、バッファ2には、常に最新のデータおよび当該データの通番を上書きし、メッセージキュー3には、上書きしたときのメッセージ31を順次メッセージキュー3の末尾につないで送信依頼することとなる。

【0031】図3は、本発明のバッファ内容例を示す。これは、図1の共用空間に設けたバッファ2の例である。ここでは、先頭アドレスaから所定サイズLの領域がバッファA（プロセスA用のバッファ2）の領域であって、図示のようにデータの通番aおよびデータを上書きする。同様に、先頭アドレスb、cから所定サイズLの領域がバッファB、C（プロセスB、C用のバッファ2）の領域であって、図示のようにデータの通番b、c

6

およびデータをそれぞれ上書きする。ここで、サイズLは任意にそれぞれ設定してもよい。

【0032】図4は、本発明のメッセージ内容例を示す。これは、図1の共用空間に設けたメッセージキュー3につなぐメッセージ31の例である。ここでは、図示の下記のような情報を設定する。

【0033】

- ・種別：例えば“処理要求”、“ステータス通知”
- ・送信元プロセスID：例えばプロセスAの場合には“A”
- ・バッファの先頭アドレス：例えばプロセスAの場合には、先頭アドレスa（図3参照）。

【0034】データの通番：例えばプロセス毎のシーケンシャル番号（1、2、3・・・）

図5は、本発明のサーバプロセスの受信処理フローチャートを示す。

【0035】図5において、S11は、メッセージを受信する。これは、図1のサーバプロセス4の受信処理41がメッセージキュー3の先頭からメッセージ31を1つ取り出す。

【0036】S12は、メッセージ31中のデータの通番Xを取得する。これは、S11で取り出したメッセージ31中のデータの通番Xを取得する（図4参照）。S13は、メッセージ31で指定されたバッファ2中のデータの通番を取得する。これは、S11で取り出したメッセージ31中に設定されていた先頭アドレスのバッファ2の位置からデータの通番Yを取得する。

【0037】S14は、X=Yか判別する。YESの場合には、メッセージ31中の通番Xとバッファ2中のデータの通番Yとが等しく最新のデータに対するメッセージ31と判明したので、S15でバッファ2中のデータを取得し、S16でこの取得したデータの処理を行なう（例えば後述する図6に示すように、画面上に表示する）。一方、S14のNOの場合には、最新のデータに対するメッセージ31でないかと判明したので、S11に戻り、メッセージキュー3から次のメッセージ31を取り出し、S12以降を繰り返す。

【0038】以上によって、サーバプロセス4中の受信処理41がメッセージキュー3の先頭からメッセージ31を取り出し、このメッセージ31に設定されていた通番と、設定されていた先頭アドレスのバッファ2のデータの通番とを比較して一致したときに最新のデータと判明したので、当該データの処理を行い、一方、一致しなかったときに古いデータと判明したのでデータの処理を中止して次のメッセージ31の受信処理に進む。これらにより、古いメッセージ31の処理を中止して次のメッセージ31の処理に進み、最新のデータの処理のみを行ない、データ処理のリアルタイム性を大幅に向上させることが可能となった。

【0039】図6は、本発明の応用例を示す。これは、

7

対象物が時間の経過に従い、個数や人数などが変化するデータをクライアントプロセス1が非同期にバッファ2およびメッセージキュー3を用いてサーバプロセス4に処理依頼し、サーバプロセス4が画面上にその個数や人数を表示する様子を模式的に表したものである。横軸が時間軸である。

【0040】図6の(a)は、実データを示す。これは、図1のクライアントプロセス1が対象物について検出した対象物の個数や人数などの変化を示す。ここでは、

- ・時刻  $t_1$  : 10
- ・時刻  $t_2$  : 15
- ・時刻  $t_3$  : 20

と変化したデータを検出したので、これらのデータを検出した時に即座にバッファ2に上書きおよび通番を設定してその毎に通番とバッファ2の先頭アドレスを設定したメッセージ31をメッセージキュー3につないで順次送信依頼する。

【0041】図6の(b)は、実時間を示す。これは、図6の(a)の実データを検出した実時間が  $t_1$ 、 $t_2$ 、 $t_3$  であることを表す。図6の(c)は、従来処理(図7のプロセスDによる処理)を示す。この従来の図7のプロセスDによる処理では、時刻  $t_1$  のデータ“10”については最初であるので、実時間と同じ時間のときに“10”として表示できる。しかし、時刻  $t_2$  のデータ“15”については、メッセージキュー3につながれていた他のメッセージ31の処理を行っていたために、 $\Delta t_2$  だけ遅れて当該“15”が図示のように表示できる。更に、時刻  $t_3$  のデータ“20”については、メッセージキュー3につながれていた他のメッセージ31の処理を行っていたために、 $\Delta t_3$  と大幅に遅れて当該“20”が図示のように表示されることとなり、特に時刻  $t_3$  のデータ“20”は、リアルタイムに表示されているとは言えなくなる。

【0042】図6の(d)は、本発明の処理(サーバプロセス4による処理)を示す。この本発明のサーバプロセス4による処理では、時刻  $t_1$  のデータ“10”については最初であるので、実時間と同じ時間のときに“10”として表示できる。続いて他のメッセージ31の処理を終わり、時刻  $t_3$  を若干経過した時点でメッセージキュー3から時刻  $t_2$  のメッセージ31を取り出した場合、バッファ2中のデータおよび通番が時刻  $t_3$  のものに上書きされているので、当該時刻  $t_2$  のデータ“1

(5)

8

5”の表示処理を中止し、次の処理に進む。そして、メッセージキュー3から時刻  $t_3$  のメッセージ31を取り出した場合、バッファ2中のデータおよび通番が時刻  $t_3$  ののものであって、一致するので、当該時刻  $t_3$  のデータ“20”の表示処理を行い、①に示すようにリアルタイム処理(表示)できる。

【0043】尚、対象物としては、入場者数、交通量、予約状況、温度、その他(数字などで監視できる物)があり、これらの対象物の数量や人数などをリアルタイムに表示などする。

【0044】

【発明の効果】以上説明したように、本発明によれば、各クライアントプロセス1がバッファ2にデータと通番を書き込むと共にメッセージ31にも通番を設定してメッセージキュー3につないで処理依頼し、サーバプロセス4がメッセージキュー3から取り出したメッセージ31の通番とバッファ2上のデータの通番とが等しい場合にのみ処理を行い、等しくないときに処理中止して最新のデータのみの処理を行なう構成を採用しているため、依頼を受けたデータの処理をリアルタイムに行なうと共にたとえメッセージキュー3に多くのメッセージ31が何らかの原因で滞留しても無駄な処理を中止して速やかにデータをリアルタイムに処理する状態に戻すことができる。

【図面の簡単な説明】

【図1】本発明のシステム構成図である。

【図2】本発明のクライアントプロセスの送信処理フローチャートである。

【図3】本発明のバッファ内容例である。

【図4】本発明のメッセージ内容例である。

【図5】本発明のサーバプロセスの受信処理フローチャートである。

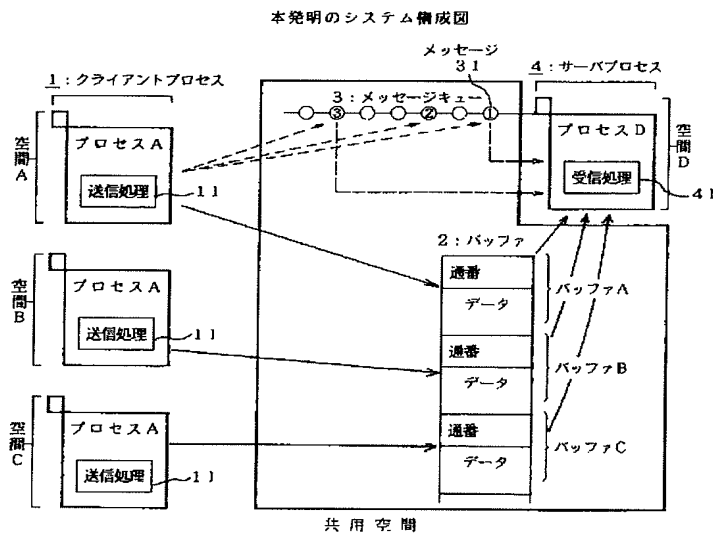
【図6】本発明の応用例である。

【図7】従来技術の説明図である。

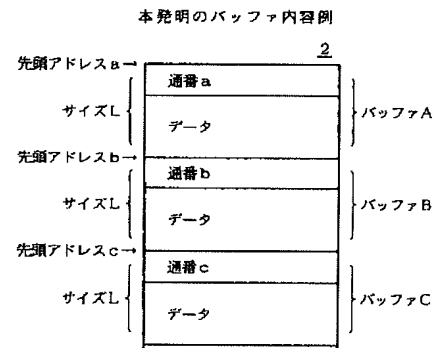
【符号の説明】

- 1 : クライアントプロセス
- 11 : 送信処理
- 2 : バッファ
- 3 : メッセージキュー
- 31 : メッセージ
- 4 : サーバプロセス
- 41 : 受信処理

【図1】

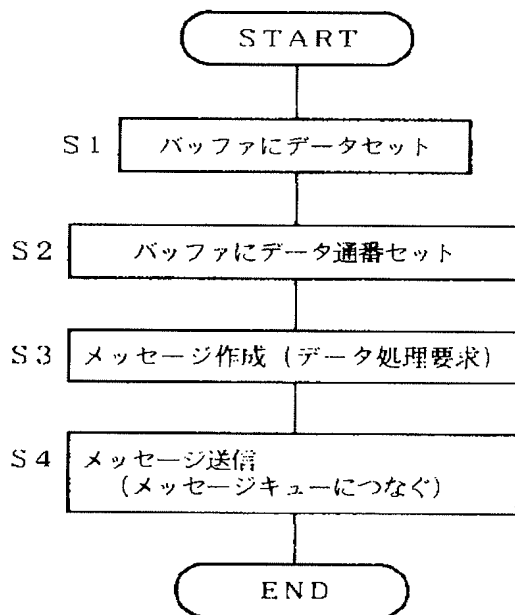


【図3】



【図2】

本発明のクライアントプロセスの送信処理フローチャート



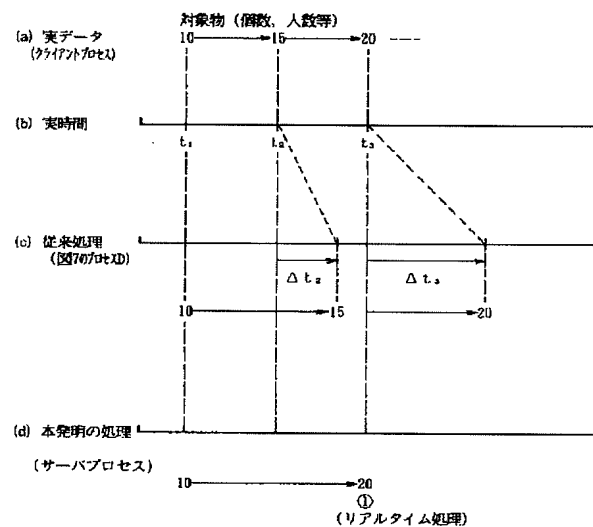
【図4】

本発明のメッセージ内容例

31			
種 別	送信元プロセスID	バッファの先頭アドレス	データの通番
・データ処理要求 (・ステータス通知)	・A		・プロセス毎の シーケンシャル番号 (1,2,3,...)

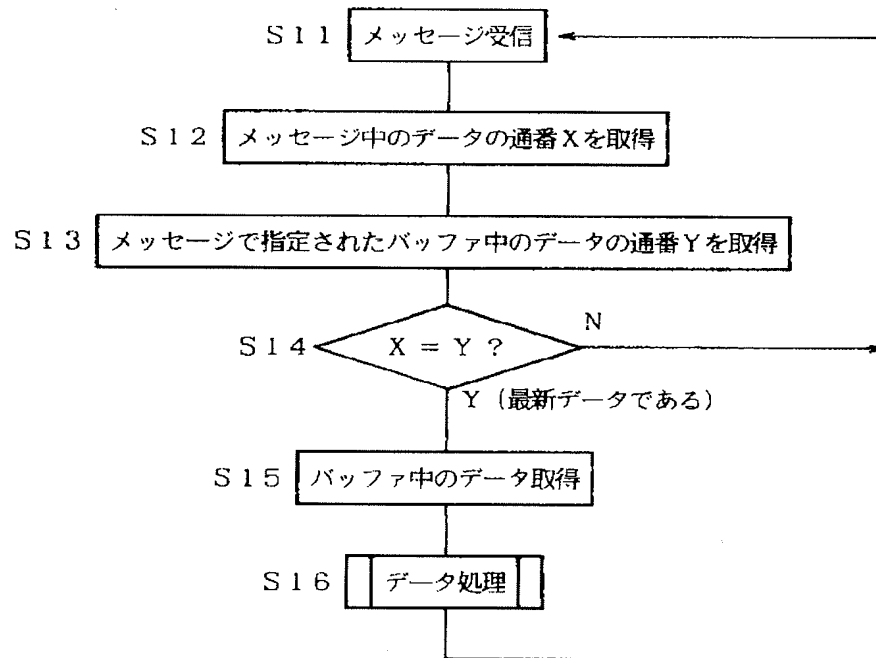
【図6】

本発明の応用例



【図5】

本発明のサーバプロセスの受信処理フローチャート





【図7】

